



Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Test data compression using Lingering Component Reduction technique for system-on-a-chip applications[☆]

K. Chakrapani*, R. Muthaiah

School of Computing, SAstra University, Thanjavur, India

ARTICLE INFO

Article history:

Received 20 February 2017

Revised 4 September 2017

Accepted 4 September 2017

Available online xxx

Keywords:

Lingering Component Reduction

Standard testing

A reference pattern Don't-care conditions

Intellectual property cores

International Symposium on Circuits and Systems'89

ABSTRACT

As technology, processes scale up, and design complexities grow, system-on-chip integration continues to rise rapidly. According to these trends, increasing test data volume is one of the largest challenges in the testing industry. In this work, we present a new test data compression method based on a stored set called Lingering Component Reduction (LCR) Code. The test set contains a large number of don't-care that can be exploited to improve the experimental data compression. In this method, a reference pattern is organized, and an adaptability of input pattern with reference pattern is reduced. The Lingering Component estimation based on test pattern contains more don't-care conditions that can be actively reduced to improve the test data compression on the Intellectual Property cores. The simulation results show the performance of experimental data compression ratio and testing time parameters. From the analysis of simulation results, it is proved that the proposed LCR code enhances a compression ratio and reduce the test time follows the International Symposium on Circuits and Systems'89. The comparative results of several benchmark circuits in maximum cases and compared the outputs of the many previous works without a substantial load on the hardware.

© 2017 Published by Elsevier Ltd.

1. Introduction

The high complexity of modern Systems-on-a-chip (SoCs) makes their testing a highly complicated task. The amount of test data quickly increases, while, in the meantime, the internal nodes of compact SoCs directed to be less accessible from the external pins. The testing issue is additionally worsened by the use of intellectual property (IP) cores. Since their formation is typically skipped from the system integrator. In such cases, no changes can be connected to the cores or their output chains, while neither programmed test pattern generation nor fault simulation tools can be used. Presently pre-computed test sets are given by the core traders, which should be connected to the cores among testing. Numerous test data compression techniques have been suggested to decrease the test data volume of unknown-structure IP cores. The existing methods are used to set the pre-computed test vectors to reduce the test data volume. To minimize both test data volume and test application time, many methods directly convert the test sets by employing many compression codes. Due to the high area overhead of the previous techniques and the inserted component condition of the last techniques, we do not consider them further in this work.

[☆] Reviews processed and recommended for publication to the Editor-in-Chief by Associate Editor Dr. S. Smys.

* Corresponding author.

E-mail address: chakrapani.sastra@gmail.com (K. Chakrapani).

A measurable compression technique that depends on the code based compression approach for the variable-test set pattern is proposed in this work. The testing is executed appropriately using selected test set, i.e., some output chains of the test set are left from the coding. Apart from the variable output chain vectors of the proposed approach, the generated reference patterns are reusable as in the sense that they can convert compatible blocks of various sizes. Two fundamental transformations are additionally displayed to enhance the measurable properties of the test set before compression. The proposed decompression design generates the decoded variable-length blocks in parallel, violating, along these lines, the test-application-time focal points that are offered by the presence of numerous scan chains in the core. Also, the decompressor is adequately designed, so their area overhead is kept low. The proposed LCR Testing model mainly based on different Test compression and decompression coding system to decrease the test application time and other metrics are concentrated in this work.

One of the primary objectives of any SOC testing is to Reduce Test Application Time (TAT).The remainder of this paper has organized as follows. Section 2 introduces the existing and related review works; Section 3 presents the proposed materials and method details. Section 4 discusses Main Procedures for LCR Code Compression. Section 5 expresses Lingering Component Reduction Code (LCR) Section 6 shows the experimental testing and results, and the paper has finally concluded in Section 7.

2. Related works

There are various techniques introduced to reduce the test time and improvement of test compression ratio for VLSI circuits. In this section, the researchers briefly discuss the latest developments in different test data compression system.

Code-based test data compressions used data compression codes to encode test 3D shapes [1]. It includes protecting the first data into models and upcoming changes in every model into an encrypted code-word to frame the compacted data. For the decompression, a decoder turns over every code-word in the compacted data back to the relating model. Run length-based codes, word reference codes, and actual codes are three primary sources of code-based test vector compressions. A strategy based on the run-length codes that the encoder possesses running off 0s using fixed length code words [2]. To decrease the duration, a system introduced the Golomb codes that encoder keeps running off 0s with variable-length code words [3,4] And [5] improved the length of the codes given to the likelihood transformation regarding the run-length.

For the most of the run-length coding preserve the successive bits of 0s and 1s might be a superior coding just the keeps running off 0s [6–9]. As of late, [10] exhibited another sample run time compression procedure that encoded 2^n has running models that are right or conversely perfect, either inside a single test data portion or over various experimental data divisions. At the point when every single acceptable model doesn't occur in the data, word reference codes can be used for the compression. An approach has given the reuse parts of the dictionary sections to upgrade the Compression Ratio (CR) [11]. Measurable coding is a part of our shorter code words into models that happen and longer ones to those that happen less as often as possible. A perfect appropriate Huffman coding that enhanced the CR [12]. There are three sections are used in testing. 1. Multiplexing, 2. Transformation and 3. Scan chain design. By employing these structures, a test bus, and Test vectors design. A novel way to deal with reducing the test time of SoC is proposed in [13], which depends on traditional neighborhood data search for testing.

The existing approach determines the problem of TAM under control, for sample, core cluster. The approach allows the system intended to improve TAM and settles on proper decisions. Another way to deal with decrease test time given the data transmission coordinating idea, which marks the multi-recurrence TAM plan. By Using the techniques for serialization and deserialization, a high data transfer capacity source and sink are associated with the small transmission capacity source and sink until transfer speed matches. In the second approach, the issue with the post-silicon approval technique, which has restricted to move speed to get to internal signs, is depicted in [14].

A TAM for the different identical cores is performed in [15], which uses the uniform way of the core and applies test at the same time, which minimizes test time for the circuit under test. In another approach, testing of the different identical core is done in a manner that, rather than taking a run of the industry test output data from the base. The TAM design presented in this work depends on the on-chip comparator and the analyzer. Reconfiguration of the numerous progress combine to reduce test application time for SoC is displayed in [16].

The run-length-based strategy codes like frequency-directed run length code (FDR) [17] Huffman code includes Selective Huffman code (SHC) [21], Optimal Selective Huffman code (OSHC) [24], and GOLOMB [25]. Variable-Length Input Huffman code (VIHC) [18] utilizes the properties that can be loaded with every one of the 0s or 1s to consolidate back to back accurate blocks. 2^n -PRL Compression technique [19] iteratively codes $2|n|$ has running of total or conversely real samples either inside an individual portion or over numerous sections as a code word. By this compression, a ratio is extremely constrained to $2|n|$ has run over more parts of outer coding. Nine coded algorithm [20] receives correct nine code(9C) words to pack test data of IP cores.

2.1. Novelty of the work

This paper shows the consideration of switching state transition with less scan chain sequences and its performance. The proposed method performs scan chain groupings and follows them to less switching state transition for each scan chain. The characteristic of the proposed sequence can be compressed as demands the following,

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات